System and Method for Error-control for Multicast Video Distribution

inventor: Jack Yiu-bun LEE

RELATED APPLICATION(S)

The present patent application is related to and claims the benefit of priority from commonly-owned U.S. Provisional Patent Application No. 60/228,772, filed on August 30, 2000, entitled "Error-Control Method for Multicast Video Distribution", which is hereby incorporated by reference in its entirety for all purposes.

BACKGROUND OF THE INVENTION

The present invention relates to networking. The present invention is especially applicable to controlling errors in the providing of content to multiple users, especially video content across broadband digital networks, especially residential broadband networks, such as cable or satellite TV networks.

With the slow but steady deployment of broadband residential networks, delivering high-quality digital video to a large number of simultaneous users becomes feasible. Examples include interactive video-on-demand and digital video multicasting or broadcasting. To compete with the conventional video delivery media (e.g. VCR tapes, LD, and DVD), these services not only have to be cost-effective and convenient, but also have to achieve competitive video quality.

Unlike the Internet, which is an extremely error-prone environment with little or no performance guarantees, broadband residential networks generally have higher bandwidth, lower latency, lower loss, and the performance is relatively stable. Nonetheless, occasional data losses in such networks are inevitable and would certainly result in video-quality degradations at the receiver end. While it is generally acceptable to receive lower-quality video over the Internet, consumers would be much less tolerant of degradation of video delivered over broadband residential networks where high-quality video is the norm (e.g. TV broadcast and

cable TV) rather than the exception.

The general problem of error control is not new and has been studied extensively over more than two decades. Traditionally, there are two major classes of error-control algorithms: retransmission-based algorithms, commonly known as Automatic Repeat Request (ARQ), and redundancy-based algorithms, commonly known as Forward Error Correction (FEC). In ARQ, lost packets are transmitted repeatedly until correctly received by the receiver. In FEC, additional redundant data are injected into the data stream so that errors or erasures can be corrected at the receiver.

SUMMARY OF THE INVENTION

What is needed is a system and a method for controlling errors that is specifically optimized and well suited to use in applications such as multicast video distribution over broadband residential networks and the like.

Some embodiments of the present invention include an efficient error-control system and method for recovering packet losses, especially losses in distributing multicast video over broadband residential networks. The system and method integrates two existing classes of error-control algorithms: Automatic Repeat Request (ARQ) and Forward Error Correction (FEC), to reduce traffic overhead and achieve scalability. Preferably, unlike most existing error-control algorithms designed for Internet multicast, the system and method does not employ substantial feedback suppression. Preferably, the system and method does not employ substantial multicasted retransmission. Preferably, the system and method does not employ substantial parity retransmission. Preferably, the system and method does not employ substantial local loss recovery.

BRIEF DESCRIPTION OF THE DRAWINGS

DETAILED DESCRIPTION OF SPECIFIC EMBODIMENTS

The description above and below and the drawings of the present document focus on one or more currently preferred embodiment(s) of the present invention and also describe some exemplary optional features and/or alternative embodiments. The description and drawings are for the purpose of illustration and not limitation. Section titles below are terse and are for convenience only.

I. First Overview: ARQ and FEC

In most applications, ARQ algorithms are more bandwidth efficient than FEC algorithms. Assuming the use of receiver-initiated selective-repeat ARQ, additional traffic overhead is incurred only when there are packet losses whereas constant traffic overhead is incurred in FEC. However, this is true only when the application is unicast-based or the number of concurrent users is small. In particular, ARQ algorithms are well known to suffer from the request-implosion problem in the context of multicast data distribution if the user population is large. This is because retransmission is performed for each receiver independently, and hence the amount of traffic overhead incurred in retransmission is proportional to the number of receivers in the multicast session. As a video server only has finite transmission (including retransmission) capacity, it will eventually be overloaded by the traffic overhead when the number of receivers grows beyond the capacity limit. While ARQ performs better for small user populations, FEC scales better for large user populations. In particular, the traffic overhead incurred in FEC is independent of the number of users in the multicast session, provided that all users have similar parameters (e.g. packet-loss probability). In short, ARQ algorithms are more bandwidth efficient for small user populations while FEC algorithms are more bandwidth efficient for large user populations.

According to an embodiment of the present invention, a hybrid system and method integrates ARQ and FEC to combine the strengths of both algorithms. The

hybrid system and method are for recovering packet losses in multicast-video distribution applications. The hybrid system and method can operate in two configurations, namely passive recovery mode and active recovery mode. The hybrid system and method can be described by performance metrics including traffic overhead, receiver buffer requirement, and server buffer requirement. These methods can be computed for the hybrid system and method, as well as for ARQ and FEC using analytical models. Using numerical results computed over a wide range of parameters, it is shown that the hybrid system and method have significant performance advantages when compared to ARQ and FEC.

II. Second Overview: Other Specific Approaches

The hybrid system and method have advantages over other specific approaches to error control, which are discussed below in this section. In general, embodiments of the hybrid system and method exist without regard to these other specific approaches. However, according to a particular set of embodiments of the hybrid system, for each such other specific approach discussed below in this section, the hybrid system and method (i) preferably does not perform or include the each such other specific approach and (ii) preferably does not perform or include any element or step of the such other specific approach, if that element or step is not specifically needed in the hybrid system. According to another embodiment of the present invention, the hybrid system and method (i) preferably does not perform or include any of the other specific approaches discussed below in this section and (ii) preferably does not perform or include any element or step of any of the other specific approaches discussed below in this section, if that element or step is not specifically needed in the hybrid system.

The problem of error recovery for multicast transmission has been studied by many researchers. Carle and Biersack [1] have done an extensive survey of error-recovery techniques for IP-based audio-visual multicast applications, covering both error recovery and error concealment techniques. Most of the studies employ

ARQ for error control and use some form of feedback suppression and/or local recovery to reduce bandwidth overhead in scaling the system to a large number of receivers. For example, the studies by Floyd *et al.* [2] used multicast to send retransmission requests so that other receivers can suppress duplicate requests to reduce overhead. Lin *et al.* [3] proposed a technique called *ACK fusion* where receivers are organized into a hierarchy. Duplicate acknowledgements (ACKs) can then be combined (*fused*) before passing up the hierarchy. These approaches indeed can reduce the amount of upstream traffic going from the receivers to the sender but at the expense of additional delay. This is undesirable for multicast video distribution as video has stringent playback deadlines. If one cater for the worst-case delay under such feedback-suppression algorithms, the additional delay incurred could lead to very large receiver buffer requirement and long response time.

Rather than simply suppressing duplicate feedback requests, Xu, *et al.* [4] went a step further to propose the use of *local loss recovery* to achieve better scalability. Specifically, receivers in a multicast session are again organized into a hierarchy such that each receiver has one or more parent nodes. A receiver sends requests to a parent node rather than the source for retransmission. If the parent node happens to have correctly received the required packets, then it could retransmit those packets to the receiver directly. Otherwise, the parent node might seek retransmission from a higher-layer parent node, and so on until the lost packets are retransmitted. Similar hierarchical schemes with local loss recovery have also been studied by Papadopoulos *et al.* [5], Costello [6] and Lucas *et al.* [7]. This local-recovery approach is particularly attractive in Internet data delivery due to the long and varying delay between the sender and the receivers.

The main disadvantage of this local-recovery approach is the need for other receivers to participate in recovering lost packets for a certain receiver. In particular, a management protocol is needed for the system to select some receivers to act as *retransmission agents* and others as *passive receivers*. Hence, the performance of

the protocol is likely to be affected if receivers frequently join and leave a multicast session. Moreover, while the protocol works well in large multicast groups, it may not work well if the multicast group is small because local recovery will be less effective. On the contrary, the hybrid algorithms studied in this paper works well for multicast groups of all sizes. Furthermore, error-control is served by the source rather than by other receivers, hence reducing the complexity of the receivers as well as the need to adapt to membership changes in the multicast session. For example, if a user abruptly shuts down a receiver (e.g. disconnecting power) that happens to be a parent node, then the dependent receivers would loose their loss-recovery support.

In another study by Nonnenmacher *et al.* [8], they studied the combination of FEC with ARQ for error-recovery in data multicast applications. In particular, they considered two approaches in combining FEC and ARQ: layered FEC and integrated FEC. The first approach is similar to the passive recovery mode in our hybrid algorithm. The second approach, called integrated FEC, retransmits redundant packets computed across multiple packets in place of the lost packet. In this way, the same retransmitted redundant packet can be used to recover different lost packets within the same parity group. They showed that their hybrid algorithm performs better than FEC. As this study focused on the delivery of discrete-media rather than continuous-media like audio and video, they did not consider the issue of media playback continuity nor receiver buffering with respect to the proposed error-recovery algorithms.

A similar approach has also been studied by Rubenstein *et al.* [9]. Their proactive forward error correction algorithm also sends redundant packets in addition to normal data packets. However, during retransmission they allow the receiver to request transmission of more-than-necessary redundant packets so that future retransmission attempts can be avoided. As proactive FEC is designed for real-time multicast over the Internet, it incorporates sophisticated algorithms to adapt the protocol to varying network conditions. By contrast, this paper focuses on

multicast video distribution over broadband residential network environments. In particular, the more predictable network environment enables us to derive a performance model for the proposed hybrid ARQ/FEC algorithm that incorporates not only the network model, but also the video playback process at the receiver as well.

In the study by Pejhan *et al.* [10], they proposed the use of multicast in sending feedback requests for negative acknowledgements (NACKs) from receivers to a server so that the receivers can learn of each other's NACKs to apply feedback-suppression schemes to reduce duplicate NACKs. Their results showed that multicasting requests can improve performance if the receivers are locally concentrated. Otherwise, the overhead in delivering multicast request packets to far-apart receivers can incur significant overhead in the network. In our hybrid algorithms, we use unicast to deliver request packets directly from receivers to the server and hence do not require the receivers to be locally concentrated. As shown in Section VIII, the integration of FEC with ARQ already substantially reduced the need for retransmission.

A different approach called Layered Video Multicast with Retransmission (LVMR) has been proposed by Li *et al.* [11]. In LVMR, a video stream is divided into layers and then multicasted on independent channels. A receiver can combine multiple layers to yield different levels of video quality. Receivers are organized into a hierarchy to reduce error-control overhead at the source and retransmissions are multicasted to further reduce duplicate requests. The distinctive feature of LVMR is its ability to adapt to changing network conditions by adding layers to or dropping layers from the current video session. This study differs from the hybrid algorithms studied in this paper in two major ways: Firstly, this protocol requires support from the video codec. In particular, the video codec must support layered encoding. Hybrid ARQ/FEC on the other hand, can work with any video codec. It is even possible to apply hybrid ARQ/FEC to individual channels of the layered video transmissions. Secondly, the need to maintain a hierarchical organization among

receivers in LVMR also poses robustness problems when receivers join, leave, or even crash during the course of a multicast session. This problem does not exist in hybrid ARQ/FEC as communications between the source and the receivers are direct.

III. Hybrid Error-Control Algorithm

In this section, we present the hybrid error-control system and methodology according to an embodiment of the present invention and explain two variants of the methodology.

FIG. 1 is a schematic block diagram that shows a broadband system 100, that when programmed or configured as discussed herein, is an embodiment of the present invention. The system 100 includes a server 110 and clients 120 coupled by a network 130, which is preferably a residential broadband network.

For the purpose of explanation and understanding, for distributing video using multicast, we assume that the video server transmits multicast video packets of size Q_S bytes periodically with a period of T_S seconds. In addition to normal video packets, the server also generates R redundant packets for every D video packets. These $(R+D)$ packets then form a parity group. For the special case of R equal to one, the redundant packet can simply be computed from the exclusive-or among the D video packets. For R larger than one, more sophisticated channel coding scheme such as Reed-Solomon codes can be used. With these R redundant packets in place, the receiver can always reconstruct all D video packets as long as any D out of these $(R+D)$ packets are correctly received. In other words, the parity group can tolerate up to R packet losses.

If there are more lost packets than can be recovered using erasure correction alone, then the receiver will request retransmission from the server. As broadband residential networks have relatively low loss rate, we can assume that packet losses among different receivers are uncorrelated. Hence, retransmission request (i.e.

NACK) and reply are both sent using unicast instead of multicast. Multicasting retransmission requests and replies are more effective for networks with higher loss rate and correlated packet losses (e.g. Internet) [8-10].

With redundant packets in the video stream, the receiver can recover lost packets by erasure correction in addition to using retransmission. In particular, the receiver can either attempt to recover lost packets first by erasure correction and then by retransmission – *passive recovery*; or first by retransmission and then by erasure correction – *active recovery*.

A. Passive Recovery

In passive recovery as depicted in FIG. 2A, the receiver attempts to recover lost packets first by erasure correction using a conventional redundancy recovery algorithm. With a redundancy of R, the receiver can recover up to R lost packets without retransmission (FIG. 2A(a)). After attempting such erasure correction, if any lost packet(s) is still not recovered, then the receiver will send a request back to the server to retransmit the lost packet(s) (i.e. NACK with selective repeat). The example in FIG. 2A(b) has one lost video packet and one lost redundant packet in a parity group. Note that only the lost video packet needs to be retransmitted because the lost redundant packet will be removed by the erasure correction process and is thereafter no longer considered missing. On the other hand, if two video packets are lost in a parity group as shown in FIG. 2A(c), both lost packets will be requested to be retransmitted.

B. Active Recovery

A shortcoming of the passive recovery mode is that correctly-received redundant packets may not be fully utilized in recovering lost packets. For example, the redundant packet in FIG. 2A(c) is not used in erasure correction (because it

cannot be done due to too many losses) but is simply discarded. To eliminate this inefficiency, we reverse the order of erasure correction and retransmission as depicted in FIG. 2B which shows the same example inputs as did FIG. 2A. Now with a redundancy of R, the receiver does not request retransmissions for *all* lost packets. In particular, only a subset of the lost packets are requested. The subset is the minimum subset that, after successful retransmission, would still leave exactly R lost packets (including the redundant packets). For example, as long as no more than R packets are lost in the parity group, all video packets can already be recovered by erasure correction (see FIG. 2B(c), which corresponds to the input of FIG. 2A(c)) and thus no retransmission is requested at all.

Unlike passive recovery, active recovery fully utilizes all received video and redundant packets for lost-packet recovery.

However, active recovery does have its own problem: the total number of lost packets is not known until the whole parity group is received. Hence, if the receiver initiates retransmission before the entire parity group has arrived, some retransmitted packets may be unnecessary if it turns out fewer than R packets are lost in this parity group. This defeats the purpose of introducing redundant packets in the first place. Therefore, the receiver defers retransmission until the whole parity group has arrived.

We have derived the performance metrics for both approaches. These metrics and numerical comparisons thereof are found in the incorporated-by-reference U.S. Provisional Patent Application No. 60/228,772. The results show that the hybrid methodology outperforms both ARQ and FEC significantly in terms of traffic overhead incurred in error recovery. Moreover, with an acceptable amount of buffering at the receiver, video playback continuity can also be guaranteed despite packet losses.

Thus, in the passive recovery mode, lost packets are first recovered by erasure correction and then by retransmission. In the active recovery mode, lost packets are retransmitted as necessary until all video packets can be recovered by

erasure correction afterward.

IV. References

G. Carle, and E.W. Biersack, "Survey of Error Recovery Techniques for IP-Based Audio-Visual Multicast Applications," *IEEE Network*, vol.11(6), Nov/Dec 1997, pp.24-36.

S. Floyd, V. Jacobson, C. Liu, S. McCanne, and L. Zhang, "A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing," *IEEE/ACM Transactions on Networking*, vol.5(6), Dec 1997, pp.784-803.

J. Lin, and S. Paul, "RMTP: A Reliable Multicast Transport Protocol," *Proc. IEEE INFOCOM '96*, San Francisco, CA, USA, March 1996.

X.R. Xu, A.C. Myers, H. Zhang, and R. Yavatkar, "Resilient Multicast Support for Continuous-Media Applications," *Proc. 7th International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV'97)*, St. Louis, MO, USA, 1997, pp.183-194.

C. Papadopoulos, G. Parulkar, and G. Varghese, "An Error Control Scheme for Large-scale Multicast Applications," *Proc. IEEE INFOCOM '98*, San Francisco, CA, USA, March 1998.

A.M. Costello, "Search Party: Using Randomcast for Reliable Multicast with Local Recovery," *Technical Report No. UCB//CSD-98-1011*, University of California Berkeley, CA, August 1998.

M.T. Lucas, B.J. Dempsey, and A.C. Weaver, "Distributed Error Recovery for Continuous Media Data in Wide-Area Multicast," *Technical Report CS95-52*, Department of Computer Science, University of Virginia, July 1995.

J. Nonnenmacher, E.W. Biersack, and D. Towsley, "Parity-Based Loss Recovery for Reliable Multicast Transmission," *In Proceedings of ACM SIGCOMM'97*, pp.289-300, Cannes, France, September, 1997.

D. Rubenstein, J. Kurose, and D. Towsley, "Real-Time Reliable Multicast Using Proactive Forward Error Correction," *Proc. IEEE NOSSDAV '98*, Cambridge, UK, July 1998.

S. Pejhan, M. Schwartz, and D. Anastassiou, "Error Control Using Retransmission Schemes in Multicast Transport Protocols for Real-Time Media," *IEEE/ACM Transactions on Networking*, vol.4(3), June 1996, pp.413-427.

X. Li, S. Paul, P. Pancha, and M. Ammar, "Layered Video Multicast with Retransmission (LVMR): Evaluation of Error Recovery Schemes," *Proc. 7th International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV'97)*, St. Louis, MO, USA, 1997, pp.161-172.

Y.B. Lee, "Concurrent Push – A Scheduling Algorithm for Push-Based Parallel Video Servers," *To appear in IEEE Transactions on Circuits and Systems for Video Technology*, Dec 1998.

S.B. Wicker, *Error Control Systems for Digital Communication and Storage*, Prentice-Hall, Englewood Cliffs, New Jersey, 1995.

H. Ohta and T. Kitami, "A Cell Loss Recover Method Using FEC in ATM Networks," *IEEE Journal of Selected Areas in Communications*, vol.9, Dec 1991, pp.1471-83.

N. Sacham and P. McKenny, "Packet Recovery in High-Speed Networks Using Coding," *Proc. INFOCOM '90*, San Francisco, CA, June 1990, pp.124-131.

Throughout the description and drawings, example embodiments are given with reference to specific configurations. One of ordinary skill in the art would appreciate that other embodiments and configurations are possible. Such other configurations are all within the scope of this invention, since with the knowledge provided by the description and drawings one of ordinary skill in the art would be able to implement the other embodiments and configurations without undue experimentation.